Computación y programación en R: Tema 1

David V. Conesa Guillén

GEeitEma

Valencia Bayesian Research group

Dept. d'Estadística i Investigació Operativa

Universitat de València

Tema 1: Introducción y preliminares.

En este tema:

1.- El entorno R. Motivos para su uso.

2.- Distribución e instalación de R.

3.- Inicio de las sesiones en R.

4.- R como calculadora científica.

5.- La ayuda en R.

6.- Salvar e iniciar sesiones anteriores.

Tema 1: Introducción y preliminares.

También:

7.- Primeras nociones: comandos, funciones y objetos.
8.- Permanencia de los datos y eliminación de objetos.
9.- Edición y ejecución de comandos desde otros programas: scripts.
10.- Librerías de `R`.
11.- Introducción a `Rcommander`.

1.- El entorno `R`. Motivos para su uso.

- `R` es, básicamente, un lenguaje que permite implementar técnicas estadísticas. Es, a la vez:
  - un entorno interactivo para el análisis estadístico y gráfico,
  - y un lenguaje de programación interpretado de alto nivel con funciones orientadas a objetos.
- El diseño de `R` vino muy influenciado por dos lenguajes existentes: `S` (Becker, Chambers and Wilks) and `Scheme` (Sussman).
- El resultado es un lenguaje de apariencia similar a `S`, pero en el que la implementación y semántica subyacente se derivan de `Scheme`.
- ¿Porqué debería cambiar a `R` cuando ya tengo bastante conocimiento de otros programas estadísticos?
- Si únicamente vamos a utilizar unos cuantos tests estadísticos, y no pensamos hacer nada más en el futuro, quizá no valga la pena el esfuerzo. PERO hay muchos motivos para aprender su uso...

1.- El entorno R. Motivos para su uso.

- Flexibilidad: R está concebido como entorno de programación con multitud de comandos y funciones específicas en estadística que permite fácilmente implementar y evaluar técnicas nuevas.
- Sacar provecho de su gran cobertura (no tiene rival) y la gran disponibilidad de aplicaciones de vanguardia en infinidad de campos.
- Entender la literatura más actual. Cada vez más gente presenta sus resultados en el contexto de R. Basta mirar cualquier disciplina y ver quien está utilizando R: la mayoría de los "popes" de cada área ya se ha cambiado. Aun más: CONTRIBUYEN y COMPARTEN.
- La distribución de R viene acompañada de un numeroso conjunto de funciones (librería `base`). Sin embargo, existen a libre disposición numerosas librerías específicas con todas las últimas técnicas disponibles (y además con explicación de su uso).
- Enorme calidad del apoyo y soporte disponible. Existe una red soberbia de "magos" de R dedicados a mejorar y a contestar las preguntas que nos salen.

1.- El entorno R. Motivos para su uso.

- El hecho que R sea un lenguaje de programación podría desanimar a muchos usuarios que piensan que no tienen "alma de programadores". Nada más lejos de la realidad:
  - R es un lenguaje interpretado (como Java) y no compilado (como C, Fortran, Pascal, etc.), es decir, los comandos escritos en el teclado son ejecutados directamente sin necesidad de construir ejecutables.
  - Además la sintaxis es simple e intuitiva.
- La estructura y facilidad de uso de R nos permite implementar nuestras propias funciones y rutinas a medida que aparecen nuestras necesidades.
- Además, también puede utilizarse para realizar gráficos de alta calidad de enorme utilidad en los trabajos de investigación.
- `Rcommander` lo convierte también en una herramienta muy útil para la docencia.
- Y, por encima de todo, es GRATIS. Uno de los mejore softwares integrados en el mundo y resulta que es nuestro por nada.

2.- Distribución e instalación de R.

- Para instalar R hay que bajar un fichero ejecutable de la página web del proyecto R:

  `http://www.r-project.org/`

- En esta página debemos hacer clic en `download CRAN`, escoger a continuación uno de los servidores (`CRAN Mirrors`) y después seguir los pasos según el sistema operativo.
- Actualmente (octubre 2017), la versión más reciente de R es la 3.4.2.

Tarea

Instalar la última versión de R tras desinstalar cualquier versión previa que pudiera haber en el ordenador de trabajo.

3.- Inicio de las sesiones en R.

- Una vez instalado, podemos ya ejecutar el programa. Para ello, sólo hace falta hacer click en el icono o desde el menú de programas seleccionar el icono correspondiente.
- Lo primero que aparece en la cabecera es la versión que estamos utilizando. Conviene visitar habitualmente la página del proyecto para ver si hay una nueva versión. Si la hay y la queremos instalar, es recomendable que desinstalemos la anterior previamente.
- El comando `demo()` permite realizar demostraciones del uso de R.
- Tal y como aparece en el primer menú, R es un software libre y viene sin GARANTIA ALGUNA, aunque es posible redistribuirlo bajo ciertas circunstancias. Más info tecleando el comando `license()`.
- Mediante `citation()` podemos saber cómo citar R o paquetes de R en publicaciones y así dar el crédito a la enorme cantidad de personas que desarrollan este proyecto.

3.- Inicio de las sesiones en R.

- En el modo por defecto, abriendo R se abre una sola ventana, la **consola** o ventana de comandos de R en la cual introduciremos los comandos y será dónde se verán los resultados de los análisis.
- Justo después de la cabecera, aparece una línea en blanco con el símbolo > en el margen izquierdo: PROMPT.
- A partir de aquí R espera que escribamos COMANDOS e instrucciones para comenzar a trabajar.
- Para ejecutar un comando escrito utilizamos la tecla de `intro`.
- El signo # indica la introducción de un comentario.
- Las órdenes elementales consisten en expresiones o en asignaciones:
  - Si una orden consiste en una expresión, se evalúa, se imprime y su valor se pierde.
  - Una asignación, por el contrario, evalúa una expresión, no la imprime y guarda su valor en una variable.

3.- Inicio de las sesiones en R.

- En cada línea sólo caben 128 caracteres: si queremos escribir más, podemos utilizar otra línea que comienza con el símbolo + en el lado izquierdo de la pantalla en vez de >. Esto también puede ocurrir cuando el comando que terminamos de escribir no está completo (falta un paréntesis, etc.).
- Si encontramos que el error es la falta de algo podemos teclearlo, y al pulsar intro, ejecutará lo que hayamos solicitado.
- Si lo que ocurre es que nos hemos equivocado, al pulsar la tecla `Esc` el prompt > volverá a aparecer.
- Con la tecla ↑ recuperamos las instrucciones utilizadas en la sesión y con las teclas → ← podemos corregir errores.
- Para separar expresiones se emplea ; (punto y coma).
- La combinación `Ctrl-C` interrumpe la edición o ejecución en curso.
- `q()` es el comando para salir de R.

4.- R como calculadora científica.

- El uso más básico de R sería utilizarlo como una calculadora científica. Así:

  ```
  3+4
  ```

  Sumará 3 y 4 y devolverá el resultado.
- Cualquier función matemática que se nos ocurra está en R:
  `log(x)`, `exp(x)`, `log(x,n)`, `log10(x)`, `sqrt(x)`, `factorial(x)`, `choose(n,x)`, `gamma(x)`, `lgamma(x)`, `floor(x)`, `ceiling(x)`, `trunc(x)`, `round(x,digits=0)`, `signif(x,digits=6)`, `cos(x)`, `sin(x)`, `tan(x)`, `acos(x)`, `asin(x)`, `atan(x)`, `acosh(x)`, `asinh(x)`, `atanh(x)`, `abs(x)`
  y podemos aplicarla sobre cualquier número.

4.- R como calculadora científica.

- Sin embargo, lo que convierte a R en una potente herramienta de trabajo es que está diseñado de forma que la mayoría de operaciones (p.e. `+,-,*,/`, etc.) y de funciones (p.e. `log()`) están definidas con carácter vectorial, es decir para operar componente a componente.
- Es conveniente pues, en la medida de lo posible, explotar dicha posibilidad ya que agiliza en mucho el tiempo de computación.
- La función principal para definir un vector es a través de sus componentes, con la función `c()`, mediante el comando más importante en R que es `<-` el de la asignación.
- Los paréntesis () se emplean para los argumentos de las funciones y para agrupar expresiones algebráicas. Los corchetes [ ] o dobles corchetes [[ ]] para seleccionar partes de un objeto así como el $. Las llaves { } para agrupar expresiones.
- Para referirnos a la componente enésima del vector `v` escribimos `v[n]`.

Tarea

Ejecutar los siguientes comandos:

`log(((3+2)*5)+6)`

Equivale a `log({(3+2)*5}+6)` o `log({{3+2}*5}+6)`

`v<-c(2,1,3,4)`

crea un vector llamado *v* con cuatro valores

`v`

nos muestra el valor de dicho vector

`w<-c(0,2,-2,1)`

crea un vector llamado *w*

`w[3]`

nos muestra la tercera componente de *w*

Tarea

Calcula las siguientes operaciones, analiza los resultados obtenidos y observa los mensajes de advertencia:

`2*v-3*w+2`

las operaciones afectan a todas las componentes

`v*w`

multiplica componente a componente

`w/v`

divide componente a componente

`v∧3`

eleva al cubo cada componente

`v∧w`

eleva cada componente de `v` a su correspondiente componente de `w`

Tarea

`sqrt(w)`
: produce un "*warning*" por pedir que haga la raíz de un negativo

`vw<-c(v,w)`
: une los dos vectores y los almacena en uno nuevo `vw`

`vw`
: muestra el resultado de `vw`

`vwa<-c(vw,6)`
: une otra componente

`vwa/v`
: da un "*warning*" pues ambos vectores no tienen el mismo número de componentes. Pero, ¿qué hace?

5.- La ayuda en R.

- R dispone de una ayuda *muy completa* sobre todas las funciones, procedimientos y elementos que configuran el lenguaje.
- Además de las opciones de menú propias de R, desde la ventana de comandos se puede acceder a información específica sobre las funciones de R con el comando `help` o mediante `?`.

  `help(log)`
  : Nos abre la ventana de ayuda sobre la función `log`.

  `? ls`
  : Hace lo mismo para la función `ls`.

Tarea

Consultar la ayuda sobre los comandos `apropos()` y `find()`. Consultar la ayuda sobre `if` (observar la diferencia entre `?if`, `help("if")`) y `help(if)`.

5.- La ayuda en R.

- Cuando no sepamos el nombre exacto del comando o de la función que necesitamos, pero sí que sepamos el tema sobre el que queremos ayuda (en inglés), podemos utilizar la función `help.search()`.
- Por ejemplo, si queremos introducir datos, y no sabemos que comandos nos pueden ayudar:

  `help.search("data input")`

  Con un poco de suerte aparecerá información sobre el nombre de funciones asociadas a esta búsqueda .

  `?read.table`

  Nos aportará dicha info.
- Es posible acceder a manuales vía el menú `Ayuda-Manuales` (en PDF) o via `CRAN`:

  `http://cran.r-project.org/`

Los manuales disponibles son:

- *An Introduction to R* introduce el lenguaje, a la vez que presenta cómo utilizar R como herramienta para el análisis estadístico y gráfico. Existe una traducción de éste en español.
- *R Language Definition* documenta el lenguaje per se, es decir los objetos en los que se basa y los detalles del proceso de evaluación. De gran utilidad cuando se va a programar funciones.
- *Writing R Extensions* cubre cómo crear nuestras propias librerías, cómo escribir los archivos de ayuda y cómo utilizar y conectar con lenguajes externos (C, C++, Fortran, etc.).
- *R Data Import/Export* describe las utilidades para importar y exportar datos en R incluyendo librerías disponibles en CRAN.
- *R Installation and Administration*.
- *The R Reference Index* (que aparece en el menú de ayuda como *R Reference manual*) contiene todos los ficheros de ayuda de la versión estándar de R junto con los de las librerías recomendadas en un único fichero imprimible.
- *R Internals* es una guía para las estructuras internas de R.

Más sobre la ayuda en R*.*

- Desde el propio programa, a través del menú `Ayuda` podemos ir a la página web de las *FAQ*. Hay dos, una más genérica **R FAQ** y otra para windows **R for Windows FAQ**.
- Si nos vemos muy apurados podemos intentar preguntar a alguien que ya haya pasado por los mismos problemas anteriormente.
- Para ello existen varias listas de correo (foros) donde preguntar dudas, aunque antes de plantear una pregunta conviene revisar antes lo que ya se ha preguntado y que por tanto ya aparece en las FAQ.
- Por supuesto, pertenecer a una lista permite recibir todas las dudas que allí en el foro se plantean (aunque a veces puede llegar a ser cansino recibir tantos correos).
- El acceso a estas listas se puede realizar desde la página web del proyecto R.

Bibliografía de R*.*

- ¡La bibliografía de R es extensísima! Imposible darla toda entera.
- En la página web del proyecto existe disponible mucha *Contributed Documentation* que además es de libre distribución.
- Entre la enorme cantidad de libros (¡posiblemente en la actualidad haya más de 100!) enteramente dedicados a R destacamos únicamente dos:
  - John M. Chambers (2008), "Software for Data Analysis: Programming with R". Springer, New York.
  - Peter Dalgaard (2008), "Introductory Statistics with R", 2nd edition. Springer.
- Una muy buen referencia es: Michael J. Crawley (2007), The R book. John Wiley and Sons, Ltd.

GEeitEm

Curiosidades de R.

- La experiencia de Ross and Robert en el diseño e implementación del lenguaje está descrito en Ihaka and Gentleman (1996), R: A Language for Data Analysis and Graphics, *Journal of Computational and Graphical Statistics*, **5**, 299-314.
- La importancia de este lenguaje es tal que existen dos congresos (`useR` -anual- y `DSC` -bianual-) enteramente dedicados a su uso.
- Hasta en el New York Times se ha escrito sobre las excelencias de `R`: `http://www.nytimes.com/2009/01/07/technology/business-computing/07program.html?_r=2`
- Desde 2009 existe una publicación *The R Journal* enteramente dedicada a artículos sobre el desarrollo y la aplicación de `R`.

Tarea

Consultar toda la información disponible en la página del proyecto.

6.- Salvar e iniciar sesiones anteriores.

- Desde el menú `Archivo>guardar área de trabajo` se puede guardar una imagen del espacio de trabajo realizado hasta ese momento. La extensión para estos archivos es `.RData`.
- Para salir de `R` podemos seleccionar `Salir` del menú `Archivo`, ejecutar la orden `q()` o cerrar directamente la ventana del programa (o la consola).
  En ese momento, `R` siempre pregunta si el usuario quiere guardar el actual espacio de trabajo (si se acaba de hacer ya no es necesario).
- Podemos retomar el espacio de trabajo de otro día abriendo desde el menú `Archivo>cargar área de trabajo` la imagen anteriormente guardada, para restaurar los objetos creados en otras sesiones; también es posible haciendo doble click sobre el fichero `.RData`.
- También es posible guardar todos los comandos de la sesión desde el menú `Archivo>Guardar Histórico`. La extensión de estos ficheros es `.RHistory`. Se pueden editar: ver `scripts` en el apartado 9.

7.- Primeras nociones: comandos, funciones y objetos.

- R es mucho más que una calculadora científica en la que aplicar unos comandos o instrucciones.
- R es un **lenguaje Orientado a Objetos**. Bajo este término se esconde la simplicidad y flexibilidad de R.
- Mientras que programas más clásicos muestran directamente los resultados de un análisis, R almacena los resultados en objetos, para ser observados o analizados posteriormente, produciendo unas salidas mínimas.
- Esto puede ser un poco extraño para el usuario, pero esta característica suele ser muy útil. De hecho, el usuario puede extraer sólo aquella parte de los resultados que le interesa.
- Sin embargo en otros programas como SAS o SPSS se obtendría de modo inmediato una salida copiosa para cualquier análisis.

Clases de objetos.

- Los *vectores* son el tipo básico de objeto en R, pero existen más tipos (los veremos de modo formal en el siguiente tema).
- Las *matrices* o, más generalmente, variables indexadas (*Arrays*) son generalizaciones multidimensionales de los vectores. De hecho, son vectores indexados por dos o más índices.
- Los *factores* sirven para representar datos categóricos.
- Las *listas* son una forma generalizada de vector en las cuales los elementos no tienen por qué ser del mismo tipo y a menudo son a su vez vectores o listas. Las listas permiten devolver los resultados de los cálculos estadísticos de un modo conveniente.
- Las hojas de datos (*data frames*) son estructuras similares a una matriz, en que cada columna puede ser de un tipo distinto a las otras.
- Las *funciones* son también objetos de R que pueden almacenarse en el espacio de trabajo, lo que permite extender las capacidades de R fácilmente.

Más sobre objetos.

- Consecuente con sus orígenes en UNIX, `R` distingue entre mayúsculas y minúsculas, de tal modo que `A` y `a` son símbolos distintos y se referirán, por tanto, a objetos distintos.
- Los nombres de los objetos pueden contener sólo letras mayúsculas o minúsculas (son distintas), junto con números y puntos (NO blancos, NO _ _ , NO %, NO $, etc.).
- Los corchetes o dobles corchetes se utilizan para seleccionar partes de un objeto así como el dólar.
- Durante una sesión de trabajo con `R` los objetos que se crean se van almacenando por su nombre.
- La función `objects()` se puede utilizar para obtener los nombres de los objetos almacenados en `R`. Es equivalente a la función `ls()`.

8.- Permanencia de los datos y eliminación de objetos.

- Ya hemos comentado que la colección de objetos almacenados en cada momento se denomina espacio de trabajo (workspace).
- Los objetos creados durante una sesión de `R` pueden almacenarse en un archivo para su uso posterior (archivos `.RData` antes comentados).
- Es posible eliminar objetos con el comando `rm()`.
- Desde el menú `Misc>remover todos los objetos` se pueden eliminar todos los objetos a la vez.

Más sobre permanencia de los datos y eliminación de objetos.

- Como hay nombres que tendemos a repetir, es recomendable utilizar un directorio de trabajo diferente para cada problema que analicemos con `R`. Es posible cambiar de directorio desde el menú `Archivo` o con el comando `setwd("C:/midir")`.
- Si no sabemos en qué directorio estamos podemos averiguarlo con el comando `getwd()`.
- A veces nos interesa ejecutar varias órdenes que tengamos almacenadas en un archivo, p.e. `ordenes.R`. Si el fichero está en el directorio de trabajo, es posible ejecutarlas dentro de una sesión de `R` con la orden `source("ordenes.R")`. Esta función también está disponible en el menú `Archivo`.

9.- Edición y ejecución de comandos desde ficheros editables: scripts.

- Para operaciones que requieran varias instrucciones consecutivas, resulta especialmente útil el trabajar con un fichero de comandos editable (`script`). El mantenimiento del código ordenado y comentado es una "buena práctica estadística".
- `R` proporciona por defecto la posibilidad de trabajar con scripts como ventanas del propio programa.
- Podemos abrir un script nuevo desde el menú `Archivo/Nuevo script`. Desde ellos, podemos ejecutar con CTRL + r los comandos de la línea en la que estamos, o el bloque de comandos que tengamos seleccionado. Conviene guardar el fichero para su posterior uso.
- Existen diferentes editores que pueden facilitar el trabajo con `R`: colores de sintaxis, completa paréntesis, etc.
  - `RStudio`, `TinnR`, `RKward`
  - `Notepad`, `(X)Emacs`
  - `RWinEdt`, `word`

RWinEdt

- RWinEdt es un editor para scripts disponible para usuarios de R en el sistema de operativo Windows.
- Requiere que el editor WinEdt (programa de pago), disponible en

  http://www.winedt.com

  esté instalado en el ordenador.
- Entonces es posible instalar la librería RWinEdt y abrir el editor mediante library(RWinEdt).
- En RWinEdt se pueden abrir y editar diferentes scripts y enviarlos a ejecución en R.
- La ventaja de este editor sobre los scripts en R, es que ofrece una serie de opciones no existentes en R, entre otras, por ejemplo, comprobar rápidamente que ningún paréntesis queda sin cerrarse o marcar, copiar y pegar columnas.

TinnR

- Otra alternativa es utilizar el editor TinnR (acrónimo de Tinn is not notepad) que es similar al anterior pero de libre distribución.
- Está disponible en http://www.sciviews.org/Tinn-R/. La versión más reciente compatible con el modo MDI de R gui es la 1.17.2.4 y el ejecutable tiene el nombre Tinn-R_1.17.2.4_setup.exe.
- Actualmente (octubre 2017) la versión más reciente (ya sólo compatible con el modo SDI) es la 2.3.7.1 y el ejecutable tiene el nombre Tinn-R_2.3.7.1_setup.exe.
- Para configurar como lanzar el código seleccionado (líneas, párrafos, etc.) desde Tinn-R hasta R debemos indicar la combinación de teclas con las que queremos que se lance el código entrando en el menú R > Hotkeys of R, por ejemplo, con CTRL + Tabulador o CTRL + r, podemos pedir que lance un bloque o una línea.

TinnR

- Las versiones más actuales de Tinn-R requiere que R gui se ejecute en modo SDI (por defecto está en modo MDI).
- Para realizar este cambio debemos abrir el programa R y luego entrar en el menú Editar > Preferencias de la interfaz gráfica..., en el que podremos realizar dicho cambio al modo SDI. Para que tenga efecto grabamos el fichero resultante.
- Si queremos que cuando acabe de ejecutar los comandos en R se quede en la ventana del programa R debemos indicarlo desde el menú de Tinn-R Options > Return focus after sending to R o bien mantener sin pulsar la pestaña correspondiente (una con una flecha azul y otra roja).
- Para utilizarlo con Windows 7 es necesario cambiar un archivo de inicialización del sistema denominado Rprofile.

Tarea

Instalar Tinn-R y prepararlo para su uso.

RStudio

- RStudio es otro editor para scripts disponible para usuarios de R en todos los sistemas operativos.
- Los autores lo consideran un entornor de desarrollo integrado que combina un interfaz muy intuitiva con herramientas de código muy potentes que permiten sacar el máximo provecho a R.
- La versión es gratuita y está disponible en

  http://www.rstudio.org/
- Como los anteriores la ventaja de este editor es que ofrece una serie de opciones no existentes en R, entre otras, por ejemplo, comprobar rápidamente que ningún paréntesis queda sin cerrarse o marcar, copiar y pegar columnas.
- Pero además nos da un listado de las variables y nos da una descripción de los bancos de datos que hemos introducido. También tiene una lista de los paquetes instalados y los gráficos realizados.

10.- Paquetes (bibliotecas o librerías) de R.

- Hasta ahora no hemos mencionado mucho la palabra estadística. Sin embargo muchas personas utilizan R como un sistema estadístico. Los autores prefieren describirlo como un entorno en el que se han implementado muchas técnicas estadísticas, tanto clásicas como modernas.
- R consta de un sistema base (donde están incluidas una enorme cantidad de técnicas estadísticas y numéricas) y de paquetes (packages) adicionales que extienden su funcionalidad.
- El hecho de distinguir entre ambos conceptos es fundamentalmente una cuestión histórica. Junto con R se incluyen varios paquetes (llamadas paquetes estándar) pero otros muchos están disponibles a través de Internet en CRAN: `http://cran.es.r-project.org/`

Tarea

Conectarse a CRAN y observar los paquetes disponibles (en octubre de 2017 está cerca de 11600 –cerca de 2500 en último año– y creciendo).

Instalación de paquetes (bibliotecas o librerías) de R.

Para instalar paquetes de R:

- Desde el menú `Paquetes > Instalar paquete(s)...`
  - Primero nos pide seleccionar el "CRAN mirror", por comodidad podemos elegir el que hay en España.
  - Luego seleccionamos ya el paquete que queremos. Si el paquete necesita otros paquetes, los instala automáticamente.
- También podemos utilizar la función `install.packages()` como en `GNU/Linux`.
- La instalación no implica que los paquetes ya puedan ser utilizados. Es necesario cargar las librerías antes de empezar a usarlas. Lo mismo ocurre con las librerías existentes en la versión local de R.

Tarea

Instalar el paquete `foreign`.

Incorporando las librerías en la sesión de R.

- Existen dos maneras de hacerlo:
  - desde el menú `Paquetes > Cargar paquete...` seleccionamos la librería que queramos,
  - o bien desde la línea de comandos utilizando la función `library()`: `library(foreign)` Carga la librería foreign
- El comando `library()` abre una ventana con información sobre las librerías (paquetes) instaladas en R.
- Para obtener más información sobre estas librerías, se puede utilizar las funciones `library` y `help` conjuntamente:

  `library(help="foreign")`
  Abre una ventana de ayuda sobre la librería foreign.

11.- Introducción a Rcommander: interfaz gráfico para R.

- Una de las librerías de R más destacadas es Rcommander (Rcmdr), que nos proporciona un interfaz gráfico, a modo de ventanas, para realizar análisis no muy complicados de una manera muy accesible.
- Es una magnífica herramienta para la docencia con R, pero también un sustitutivo de paquetes tradicionales tales como SPSS, etc.
- La versión de Rcommander última (octubre 2017) es la 2.4-x.
- Se instala como el resto de librerías. Al instalarla, solicita permiso para instalar un considerable número de librerías.
- Tras instalarla, podemos cargarla de la manera habitual para poder utilizarla. Observaremos que abre una ventana con menús similar a otros programas de estadística.
- Se le pueden incorporar una serie de complementos (o plugins) que se instalan como paquetes de R. Se cargan desde el menú *Herramientas -> Cargar plugins(s) de Rcmdr*. Hay que reinicializar Rcommander para poder utilizarlos, y al hacerlo algunos menús de Rcommander cambian.

Menús disponible en Rcommander.

- En el Menú de la ventana de `Rcommander` aparece el acceso a las siguientes utilidades:
  - **Fichero**: Abrir, Guardar instrucciones, Guardar resultados, Guardar entorno de trabajo de `R`, Salir.
  - **Editar**: Limpiar ventana, Cortar, Copiar, Pegar, Borrar, Buscar, Seleccionar todo.
  - **Datos**: Nuevo conjunto de datos, Cargar conjunto de datos, Importar datos, Conjunto de datos en paquetes, Conjunto de datos activo, Modificar variables del conjunto de datos activo.
  - **Estadísticos**: Resúmenes, Tablas de contingencia, Medias, Proporciones, Varianzas, Tests no paramétricos, Análisis dimensional, Ajuste de modelos.
  - **Gráficas**: diversos gráficos univariantes, bivariantes y 3D, Guardar gráfico en fichero.

Más sobre los menús disponibles en Rcommander.

- **Modelos**: Selecciona el modelo activo, Resumir el modelo, Añadir las estadísticas de las observaciones a los datos, Intervalos de confianza, AIC, BIC, Test de hipótesis, Diagnósticos numéricos, Gráficas.
- **Distribuciones**: Distribuciones continuas, Distribuciones discretas, Visualizar distribuciones.
- **Ts-Data y Ts-Models**: análisis de datos y gráficas series temporales.
- **Herramientas**: Cargar paquete(s), Cargar plugins de Rcmdr, Opciones.
- **Ayuda**: Ayuda de R Commander, Introducción de R Commander, Ayuda conjunto de datos activo (si existe), Información sobre Rcmdr.

Más sobre Rcommander.

- Bajo los menús hay una "barra de herramientas" con una fila de botones.
  - El botón horizontal que se encuentra más a la izquierda muestra el nombre de la serie de datos activa. Inicialmente no hay ninguna serie de datos activa. Al presionar este botón, se puede elegir entre las series de datos que están actualmente en la memoria (si hay más de una). La mayoría de los menús y diálogos en `R Commander` se refieren a la serie de datos activa (los menús Fichero, Editar y Distribuciones son excepciones).
  - Los dos botones contiguos nos permiten abrir el editor de datos de `R` para modificar la serie de datos activa o el visor para examinarla. El visor de la serie de datos puede permanecer abierto mientras se realizan otras operaciones.
  - Un botón horizontal indica el nombre del modelo estadístico activo, un modelo lineal (como el modelo de regresión lineal), un modelo lineal generalizado, un modelo logit multinominal o un modelo lineal de regresión. Inicialmente no hay ningún modelo activo. Si hay más de un modelo en la memoria, puede elegir entre ellos pulsando sobre el botón.

Más sobre Rcommander.

- Inmediatamente debajo de la barra de herramientas se encuentra la `ventana de instrucciones`, una ventana que actúa como un script y en la que se copian las instrucciones generadas por la interfaz gráfica.
- Podemos editar el texto en la ventana de instrucciones o incluso escribir comandos de `R`.
- Presionando el botón Ejecutar, que está a la derecha debajo de la ventana de instrucciones (o, alternativamente, la combinación de teclas Ctrl-r para "ejecutar"), la línea contenida en el cursor se ejecuta (o se re-ejecuta) para su resolución.
- Si seleccionamos varias líneas (e.g., pulsando con el botón izquierdo y arrastrando el ratón sobre ellas), el presionar Ejecutar se ejecutan todas ellas. La combinación de teclas `Ctrl-a` selecciona todo el texto de la ventana de instrucciones, y `Ctrl-s` abre una ventana de diálogo para guardar los contenidos de la ventana.

Más sobre Rcommander.

- Bajo la ventana de instrucciones hay una ventana grande para los resultados que permite editar el texto y desplazarse arriba y abajo sobre él.
- Las instrucciones escritas en esta ventana aparecen en rojo y los resultados en azul oscuro (como en la consola de R).
- Al final hay una pequeña ventana de texto gris para mensajes. Los mensajes de error se muestran en rojo, los avisos en verde y otros mensajes en azul oscuro.
- Podemos reajustar o maximizar la ventana Rcommander de la manera usual. Si reajustamos Rcommander, el ancho de los resultados R subsiguientes automáticamente se ajustará para encajar en la ventana de resultados.
- Es posible cambiar la configuración de Rcommander a través del menú Herramientas. Ver la ayuda para más detalles.

Licencia de este material